LAMPIRAN

Program Alat

```
#include <Firebase.h>
#include <Keypad.h>
#include <ESP32Servo.h>
#include <Preferences.h> // Library untuk penyimpanan di ESP32

// Inisialisasi Preferences untuk menyimpan password
Preferences preferences;

#define WIFI_SSID "pengguna" //ganti dengan nama hotspot mu
#define WIFI_PASSWORD "220280149" //ganti dengan password hotspot mu
#define REFERENCE_URL "https://smartkeyhome-d8928-default-rtdb.firebaseio.com/" //ganti dengan URLmu
#define AUTH_TOKEN "COeBhNCTzT3NuZWXyolq11c0j326GQYydi67LPp7" //ganti dengan database secret/rahasia database mu

Firebase fb(REFERENCE_URL, AUTH_TOKEN);

// Pin untuk servo dan sudut awal
const int servoPin = 18; // Sesuaikan dengan pin ESP32 yang digunakan
const int bukaSudut = 90; // Sudut untuk membuka pintu
const int tutupSudut = 0; // Sudut untuk menutup pintu

const int buzzerPin = 4;

// Konfigurasi keypad
const byte ROWS = 4; // Jumlah baris pada keypad
const byte COLS = 4; // Jumlah kolom pada keypad

char keys[ROWS][COLS] = {
  {'D', 'C', 'B', 'A'},
  {'#', '9', '6', '3'},
  {'0', '8', '5', '2'},
  {'*', '7', '4', '1'},
};

byte rowPins[ROWS] = {13, 12, 14,
```

```
    27}; // Pin kolom keypad yang
    terhubung ke ESP32
byte colPins[COLS] = {26, 25, 33,
    32};  // Pin kolom keypad
    yang terhubung ke ESP32

Keypad keypad =
    Keypad(makeKeymap(keys),
    rowPins, colPins, ROWS,
    COLS);

// Pin untuk solenoid
const int solenoidPin = 15;

// Pin untuk Magnetic Door
const int magneticPin = 2;
String statusPintu;

// Durasi solenoid aktif dalam
    milidetik
const int unlockDuration = 5000;

// Variabel untuk password
String password ; // Password yang
    benar
String inputPassword = ""; //
    Menyimpan input dari
    pengguna

Servo servo;

void buzzer() {
  digitalWrite(buzzerPin, HIGH);
  delay(100);
  digitalWrite(buzzerPin, LOW);
  delay(100);
  digitalWrite(buzzerPin, HIGH);
  delay(100);
  digitalWrite(buzzerPin, LOW);
}

void setupWiFi() {
  Serial.begin(115200);
  WiFi.mode(WIFI_STA);
  WiFi.disconnect();
  delay(1000);

  Serial.println();
  Serial.print("Connecting to: ");
  Serial.println(WIFI_SSID);
        WiFi.begin(WIFI_SSID,
    WIFI_PASSWORD);

     while (WiFi.status() !=
    WL_CONNECTED) {
   Serial.print("-");
   delay(500);
  }

  Serial.println();
  Serial.println("WiFi Connected");
  Serial.println();
```

```
}

void taskKeypad(void *pvParameters) {
  while (true) {
    char key = keypad.getKey();
    if (key) {
      Serial.print("Input password: ");
      Serial.println(key);
      digitalWrite(buzzerPin,HIGH);
      delay(100);
      digitalWrite(buzzerPin,LOW);
      if (key == '#') {
          // Periksa apakah password benar
        if (inputPassword == password) {
          Serial.println("  Password BENAR!");
          buzzer();
            Serial.println("  Membuka kunci...");
              digitalWrite(solenoidPin, HIGH); // Aktifkan solenoid lewat relay
            servo.write(bukaSudut);
          // servo buka pintu
          fb.setString("perintah", "");
          delay(unlockDuration);
        // Tunggu selama durasi unlock
              digitalWrite(solenoidPin, LOW); // Matikan solenoid lewat relay
        } else {
          Serial.println("+ Password SALAH!");
              digitalWrite(buzzerPin, HIGH);
          delay(500);
              digitalWrite(buzzerPin, LOW);
        }
          inputPassword = ""; // Reset input password setelah tombol # ditekan
      } else if (key == '*') {
              if (inputPassword == password) {
          Serial.println("  Password BENAR!");
            Serial.println("Masukkan password baru:");
              digitalWrite(buzzerPin, HIGH);
          delay(500);
              digitalWrite(buzzerPin, LOW);
            inputPassword = "*"; // Tandai mode ubah password
        } else {
          Serial.println("+  Password
```

```
SALAH!");
            digitalWrite(buzzerPin,
HIGH);
  delay(500);
            digitalWrite(buzzerPin,
LOW);
   inputPassword = "";
 }
}else if (key == 'C') {
            Serial.println("  Cancel
Password...");
            digitalWrite(buzzerPin,
HIGH);
  delay(500);
            digitalWrite(buzzerPin,
LOW);
 inputPassword = "";
} else if (key == 'D') { // Fitur
tambahan: Menutup pintu dari
keypad
    Serial.println("  Menutup
pintu...");
      servo.write(tutupSudut);  //
Kembali ke posisi tertutup
         digitalWrite(solenoidPin,
LOW); // Matikan solenoid jika
masih aktif
     fb.setString("status",  "Pintu
Tertutup"); // Update status ke
Firebase
buzzer();
 }else {
        // Tambahkan input ke
  password
  inputPassword += key;
 }
}

   // Simpan password baru jika
   panjang cukup dan diawali
   dengan "*"
if (inputPassword.length() > 4 &&
 inputPassword[0] == '*') {
 buzzer();
        String newPassword =
  inputPassword.substring(1);

  preferences.putString("passwor
 d",  newPassword);  //  Simpan
 password baru ke memori
 password = newPassword;
   Serial.println("\n  Password
  berhasil diubah!");
 buzzer();
   inputPassword  =  "";  //  Reset
  input  setelah  perubahan
  password
 }

   delay(50);  //  Agar  tidak
  membebani CPU
}
```

```
}

void taskFirebase(void *pvParameters) {
  while (true) {
    int readMagnetic = digitalRead(magneticPin);
    if (readMagnetic == 0) {
      statusPintu = "Pintu Terbuka";
    } else {
      statusPintu = "Pintu Tertutup";
    }
    Serial.println(statusPintu);
    fb.setString("status", statusPintu); // Update status pintu ke Firebase
    String perintah = fb.getString("perintah");
    if (perintah == "tutup"){servo.write(tutupSudut);}
    else if (perintah == "buka"){
      Serial.println(" Perintah BUKA!");
      buzzer();
      Serial.println(" Membuka kunci...");
      digitalWrite(solenoidPin, HIGH); // Aktifkan solenoid lewat relay
      servo.write(bukaSudut); // servo buka pintu
      fb.setString("perintah", "");
      delay(unlockDuration);
      // Tunggu selama durasi unlock
      digitalWrite(solenoidPin, LOW); // Matikan solenoid lewat relay
      fb.setString("status", statusPintu);
    }

    delay(5000); // Kirim status setiap 5 detik
  }
}

void setup() {
  setupWiFi();

  // Setup servo dan pin lainnya
  servo.attach(servoPin);
  servo.write(tutupSudut); // Mulai dari posisi tertutup
  pinMode(solenoidPin, OUTPUT);
  pinMode(buzzerPin, OUTPUT);
  pinMode(magneticPin, INPUT);
  digitalWrite(solenoidPin, LOW); // Pastikan solenoid dalam keadaan mati saat memulai

  preferences.begin("storage", false);
```

```
    // Mulai Preferences storage

  // Membaca password dari memori
     jika ada
   if  (preferences.isKey("password"))
     {
                    password =
     preferences.getString("passwor
     d", "1234");
  }

     Serial.println("Keypad dengan
     Password siap!");



     xTaskCreatePinnedToCore(task
     Keypad, "Keypad Task", 10000,
     NULL, 1, NULL, 0);

     xTaskCreatePinnedToCore(task
     Firebase, "Firebase Task",
     10000, NULL, 1, NULL, 1);
}

void loop() {
  delay(1000);
}
```

Program Aplikasi Android

```
<!DOCTYPE html>
<html lang="id">
<head>
   <meta charset="UTF-8">
        <meta name="viewport"
     content="width=device-width,
     initial-scale=1.0">
        <title>Kontrol Pintu -
     SmartHome</title>
   <script type="module">
      import { initializeApp } from
     "https://www.gstatic.com/fireba
     sejs/9.0.0/firebase-app.js";
     import { getDatabase, ref, set,
     onValue } from
     "https://www.gstatic.com/fireba
     sejs/9.0.0/firebase-database.js";

     const firebaseConfig = {
                    databaseURL:
     "https://smartkeyhome-d8928-
     default-rtdb.firebaseio.com",
     };

                const app =
     initializeApp(firebaseConfig);
            const database =
     getDatabase(app);
```

```
function ubahData(dataBaru) {
      const dbRef = ref(database,
'perintah');
   set(dbRef, dataBaru)
      .then(() => {
           alert(' Data berhasil
diubah!');
      })
      .catch((error) => {
                alert('+ Terjadi
kesalahan: ' + error.message);
      });
}

  // Mendeteksi perubahan data
secara real-time
 const statusRef = ref(database,
'status');
  onValue(statusRef, (snapshot)
=> {

document.getElementById('fire
baseData').innerText =
snapshot.exists() ?
snapshot.val() : ▮ . Tidak ada
data';
});

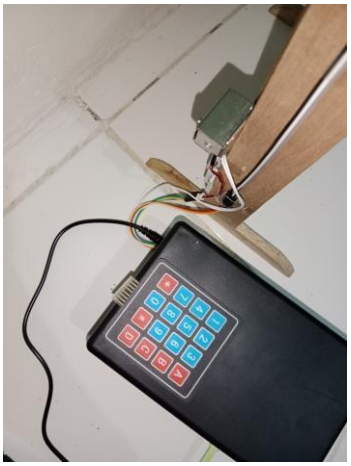
window.ubahData = ubahData;
</script>
<style>
                    @import
   url('https://fonts.googleapis.com
   /css2?family=Raleway:wght@3
   00;400;700&display=swap');

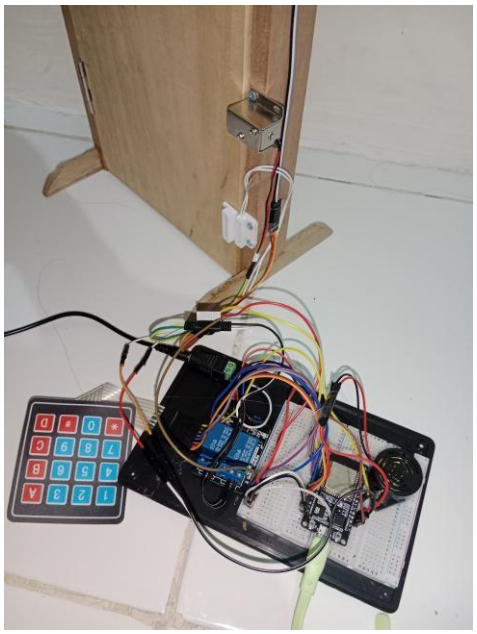
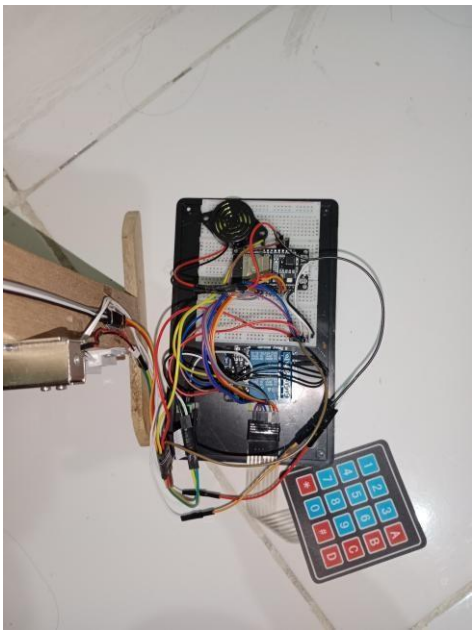
   body {
      font-family: 'Raleway', sans-
   serif;
      display: flex;
      flex-direction: column;
      align-items: center;
      justify-content: center;
      height: 100vh;
      margin: 0;
      background: #f3e5ab;
      color: #4e342e;
      text-align: center;
      padding: 10px;
   }
   .container {
      background: #d7ccc8;
      padding: 20px;
      border-radius: 12px;
        box-shadow: 0px 4px 10px
   rgba(0, 0, 0, 0.2);
      width: 80%;
      max-width: 300px;
   }
   .header {
```

```
    font-size: 20px;
    font-weight: 700;
    margin-bottom: 15px;
    color: #3e2723;
}
button {
                background: linear-
gradient(135deg, #8d6e63,
#6d4c41);
    color: white;
    border: none;
    padding: 10px 18px;
    font-size: 14px;
    border-radius: 25px;
    cursor: pointer;
        transition: transform 0.2s,
box-shadow 0.2s;
    width: 100%;
    font-weight: bold;
    display: flex;
    align-items: center;
    justify-content: center;
    gap: 8px;
}
button:hover {
    transform: scale(1.03);
        box-shadow: 0px 3px 8px
rgba(0, 0, 0, 0.2);
}
#firebaseData {
    font-size: 16px;
    font-weight: bold;
    margin-top: 10px;
    background: #a1887f;
    padding: 10px;
    border-radius: 8px;
    color: white;
        box-shadow: 0px 2px 6px
rgba(0, 0, 0, 0.2);
}
.status-box {
    margin-top: 15px;
    padding: 12px;
    border-radius: 8px;
    background: #bcaaa4;
    color: #3e2723;
    font-weight: bold;
        box-shadow: 0px 2px 6px
rgba(0, 0, 0, 0.2);
}
  </style>
</head>
<body>
  <div class="container">
          <div  class="header">
  Welcome to Smart Door</div>
                        <button
  onclick="ubahData('tutup')">
                        TUTUP
  PINTU</button><br>
                        <button
  onclick="ubahData('buka')">
```

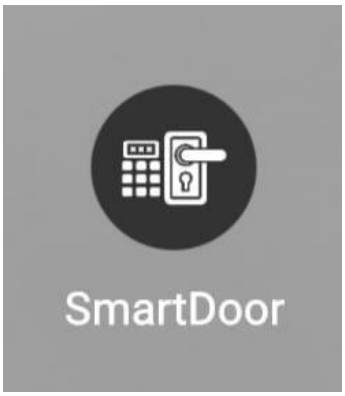
```
 BUKA PINTU</button>
<div class="status-box">
  <h2> Status Pintu</h2>
    <p id="firebaseData">
Memuat data...</p>
                                </div>
                              </div>
                            </body>
                            </html>
```

Dokumentasi Alat

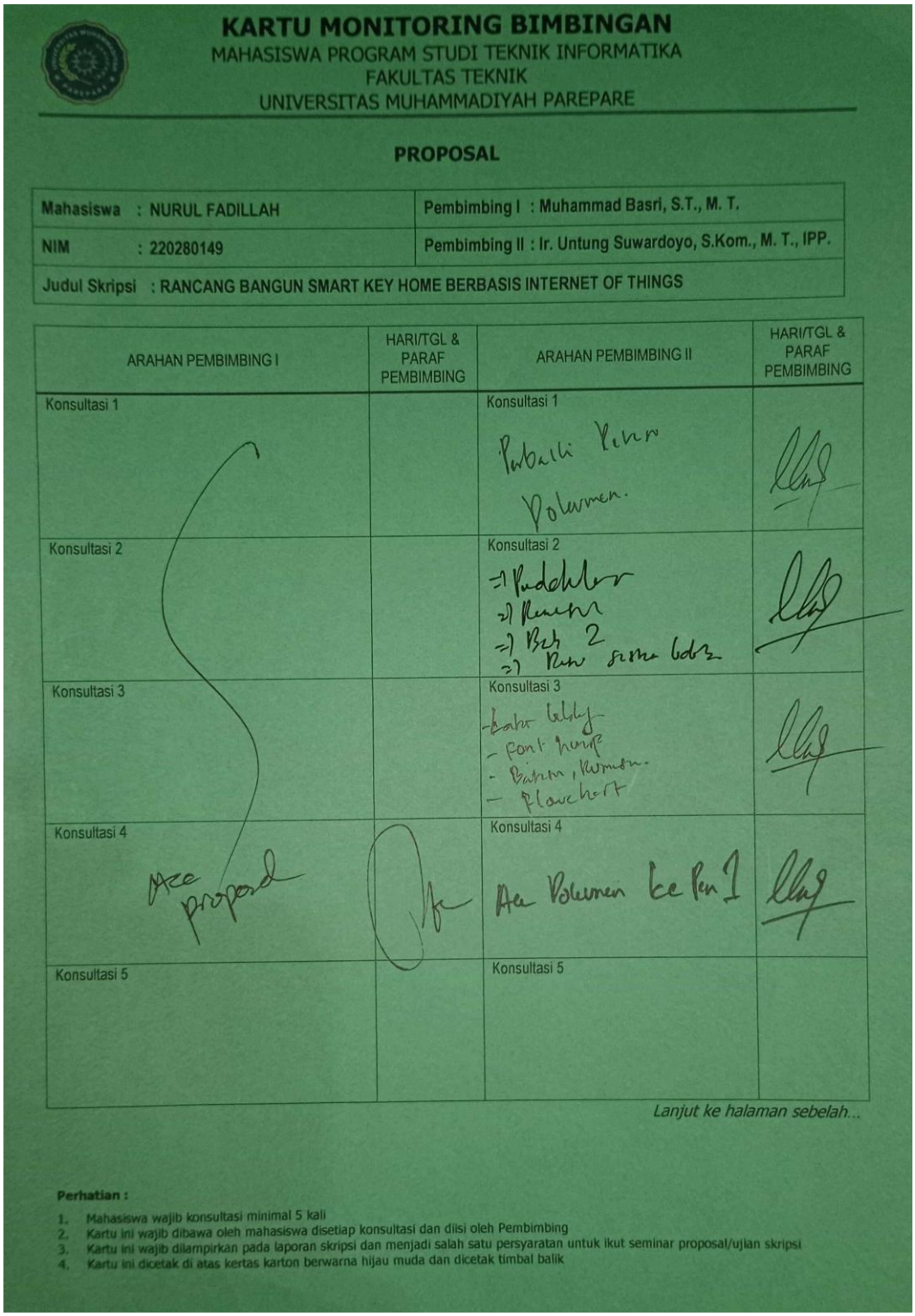



Tampilan Aplikasi Android





Kartu Monitoring Proposal



KARTU MONITORING BIMBINGAN

MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

PROPOSAL

Mahasiswa : NURUL FADILLAH	**Pembimbing I : Muhammad Basri, S.T., M. T.**
NIM : 220280149	**Pembimbing II : Ir. Untung Suwardoyo, S.Kom., M. T., IPP.**
Judul Skripsi : RANCANG BANGUN SMART KEY HOME BERBASIS INTERNET OF THINGS	

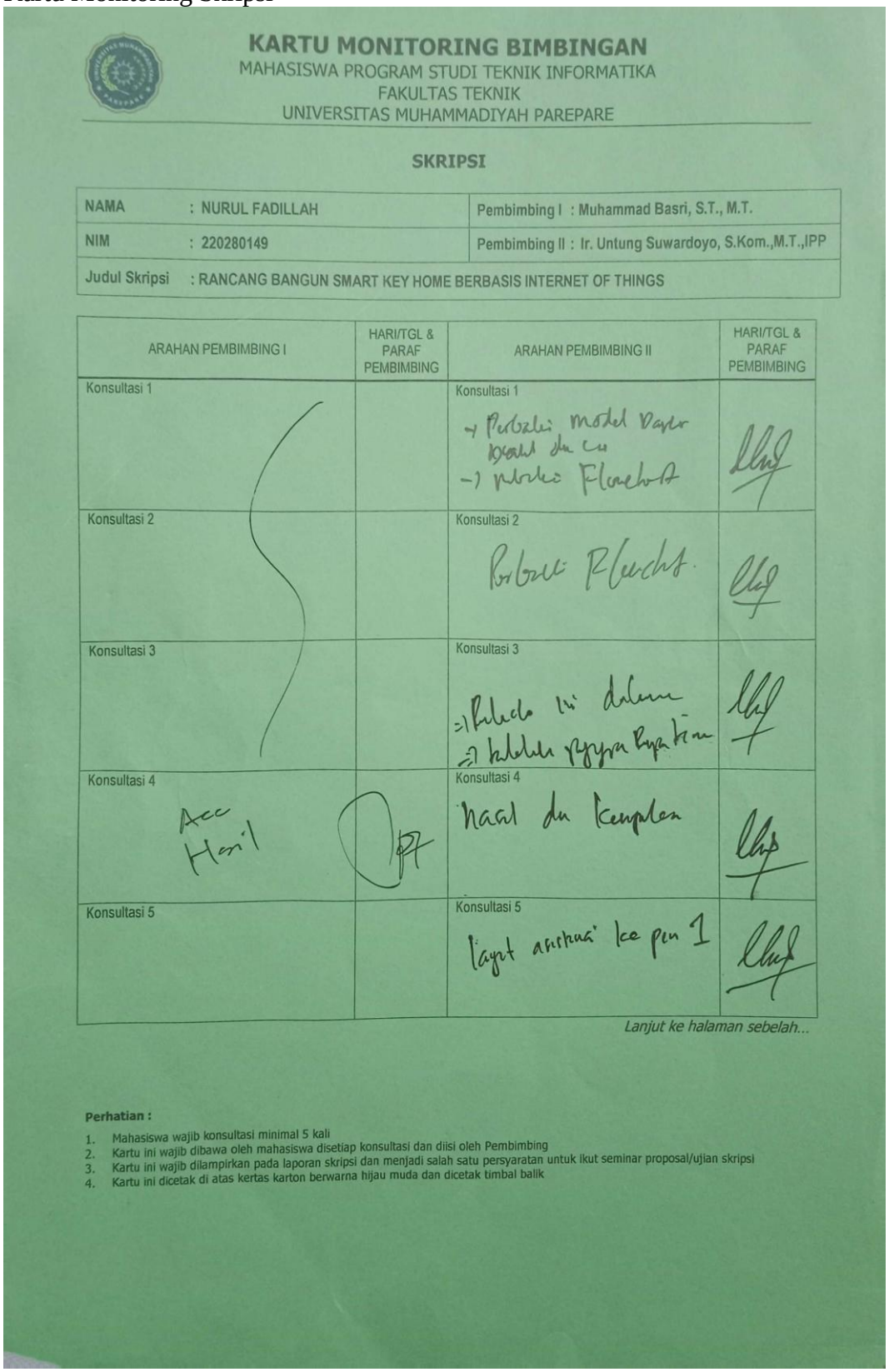
ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1		Konsultasi 1	
Konsultasi 2		Konsultasi 2	
Konsultasi 3		Konsultasi 3	
Konsultasi 4		Konsultasi 4	
Konsultasi 5		Konsultasi 5	

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik

Kartu Monitoring Skripsi



KARTU MONITORING BIMBINGAN
MAHASISWA PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH PAREPARE

SKRIPSI

NAMA	: NURUL FADILLAH	Pembimbing I	: Muhammad Basri, S.T., M.T.
NIM	: 220280149	Pembimbing II	: Ir. Untung Suwardoyo, S.Kom.,M.T.,IPP
Judul Skripsi	: RANCANG BANGUN SMART KEY HOME BERBASIS INTERNET OF THINGS		

ARAHAN PEMBIMBING I	HARI/TGL & PARAF PEMBIMBING	ARAHAN PEMBIMBING II	HARI/TGL & PARAF PEMBIMBING
Konsultasi 1		Konsultasi 1 - Perbaiki model Data Flow Diagram dan Use Case - Blok Flowchart	
Konsultasi 2		Konsultasi 2 Perbaiki Flowchart.	
Konsultasi 3		Konsultasi 3 =) Perbaiki lagi dalam =) Lebih pahami Pyton	
Konsultasi 4 Acc Hasil		Konsultasi 4 Hasil dan Kemplen	
Konsultasi 5		Konsultasi 5 Lanjut ujian ke pen 1	

Lanjut ke halaman sebelah...

Perhatian :

1. Mahasiswa wajib konsultasi minimal 5 kali
2. Kartu ini wajib dibawa oleh mahasiswa disetiap konsultasi dan diisi oleh Pembimbing
3. Kartu ini wajib dilampirkan pada laporan skripsi dan menjadi salah satu persyaratan untuk ikut seminar proposal/ujian skripsi
4. Kartu ini dicetak di atas kertas karton berwarna hijau muda dan dicetak timbal balik